Sales Forecasting for Retail Business using XGBoost Algorithm and TimesFM

Miss Prathana Dankorpho

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in Computer Science

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2024

การพยากรณ์ยอดขายโดยใช้อัลกอริทึม XGBoost และ TimesFM

นางสาวปรารถนา ด่านก่อโพธิ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2567

Thesis Title	Sales Forecasting for Retail Business using XGBoost Algorithm and TimesFM
By	Miss Prathana Dankorpho
Field of Study	Computer Science
Thesis Advisor	Professor Prabhas Chongstitvatana, Ph.D.

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirement for the Master of Science

..... Dean of the Faculty of Engineering
(Associate Professor Witaya Wannasuphoprasit, Ph.D.)

THESIS COMMITTEE

..... Chairman
(Professor Wiwat Vatanawood, Ph.D.)

..... Thesis Advisor
(Professor Prabhas Chongstitvatana, Ph.D.)

..... External Examiner
(Associate Professor Worasait Suwannik, Ph.D.)

ปรารถนา ด่านก่อโพธิ์ : การพยากรณ์ยอดขายโดยใช้อัลกอริทึม XGBoost และ TimesFM. (Sales Forecasting for Retail Business using XGBoost Algorithm and TimesFM) อ.ที่ปรึกษาหลัก : ศ. ดร.ประภาส จงสถิตย์วัฒนา

ธุรกิจค้าปลีกมีการพัฒนาอย่างต่อเนื่องทั้งในด้านการขยายช่องทางการขายและความหลากหลายของประเภทผลิตภัณฑ์ อย่างไรก็ตามวิธีการพยากรณ์ในปัจจุบันนั้นอาศัยวิธการทางสถิติพื้นฐาน ซึ่งไม่สอดคล้องกับความรวดเร็วในการเปลี่ยนแปลงสภาพแวดล้อมของธุรกิจ ข้อจำกัดนี้นำไปสู่ความท้าทายของการคาดการณ์ยอดขายอย่างถูกต้องแม่นยำ และมีความถี่ในการทำนายอย่างเหมาะสม ทันกับการเปลี่ยนแปลง ดังนั้นจึงมีความจำเป็นอย่างยิ่งที่จะต้องปรับปรุงความแม่นยำและความถี่ของการคาดการณ์การขายเพื่อให้สามารถตัดสินใจกลยุทธ์ทางธุรกิจได้ทันท่วงที จากการทดสอบด้วยชุดข้อมูลตั้งแต่ปี 2562 ถึง 2566 งานวิจัยนี้แสดงให้เห็นถึงข้อได้เปรียบที่สำคัญของการใช้ eXtreme Gradient Boosting (XGBoost) และ TimesFM เพื่อทำนายยอดขาย โดยผลลัพธ์ที่ได้แสดงให้เห็นถึงความถูกต้องที่เพิ่มขึ้นอย่างมีนัยสำคัญเมื่อเปรียบเทียบกับวิธีการเดิม นอกจากนี้การนำ XGBoost และ TimesFM มาใช้ยังทำให้ความถี่ในการทำนายยอดขายพัฒนาขึ้นจากรายเดือนเป็นรายวัน ด้วยข้อดีที่เกิดจากการพัฒนาเครื่องมือเพื่อใช้ในการทำนายยอดขายเหล่านี้ ทำให้กิจการสามารถเพิ่มประสิทธิภาพการจัดการสินค้าคงคลัง กำหนดกลยุทธ์การตลาดที่มีประสิทธิภาพ และเพิ่มรายได้ ส่งผลให้ธุรกิจเติบโตอย่างยั่งยืน

สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่อนิสิต .....

ปีการศึกษา 2567 ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

6372071921 : MAJOR COMPUTER SCIENCE

KEYWORD: eXtreme Gradient Boosting, Machine learning, Nonlinear data, Retail business, Sales Forecasting, XGBoost

Prathana Dankorpho : Sales Forecasting for Retail Business using XGBoost Algorithm and TimesFM. Advisor: Prof. Prabhas Chongstitvatana, Ph.D.

The retail industry is continuously evolving with the expansion of sales channels and the diversification of product assortments. However, current forecasting methods, relying on simplistic statistical models, frequently encounter difficulties in adjusting to the dynamic environment. This limitation leads to challenges in accurately predicting sales. Consequently, there is a critical need to improve the accuracy and frequency of sales predictions to enable timely decision-making for business strategies. Through a comprehensive analysis of datasets from 2019 to 2023, this study illustrates the advantages of integrating XGBoost and TimesFM to gain deeper insights into sales patterns. Results demonstrate a significant enhancement in prediction accuracy compared to original methods. Furthermore, the adoption of XGBoost and TimesFM facilitates the transition from monthly to daily forecasting, thereby optimizing the efficiency of the prediction process. Retailer can optimize inventory management, effective marketing strategies, and maximize revenue. The findings emphasize the importance of embracing innovative approaches to address the challenges of a rapidly evolving retail landscape and drive sustainable growth.

Field of Study:	Computer Science	Student's Signature .....
Academic Year:	2024	Advisor's Signature .....

ACKNOWLEDGEMENTS

I wish to extend my heartfelt gratitude to the individuals and organizations whose contributions were indispensable to the completion of this research paper.

First and foremost, I am deeply thankful to my supervisor, Prof. Prabhas Chongstitvatana, Ph.D., whose guidance and support were instrumental throughout this research endeavor. His feedback and encouragement significantly shaped the direction of this paper.

I am also indebted to my chairman and research committee, Prof. Wiwat Vatanawood, Ph.D., and Assoc. Prof. Worasait Suwannik, Ph.D., for their constructive comments and suggestions. Their expertise and perspectives enhanced the quality of this work.

Special thanks are extended to Central Trading Co., Ltd., for generously providing the dataset essential for our analysis during the research process.

Furthermore, I am grateful to my family, colleagues, and friends whose encouragement and moral support were a constant source of motivation during this journey.

In conclusion, the completion of this paper would not have been possible without the unwavering support of these individuals and organizations. Thank you all sincerely.

Prathana Dankorpho

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

Page

Chapter 1

Introduction

1.1 Statement of the problems

Retail businesses are growing incessantly, and there are various channels where customers can reach the products other than offline channels as in the past but expanded to online channels such as own website, e-commerce platforms, or social media. Thus, forecasting sales becomes challenging because of diverse variables and continual changes in sales that are generated from all channels. The current processes have been done by basic statistics, straight-line average by humans which is defined in this paper as the Original Method. Thus, it takes a longer time to make predictions compared to computing from machine learning. Moreover, the Original Method can provide only monthly forecasts, limiting the ability of retailers to respond promptly to changing market dynamics and consumer behaviour.

1.2 Objective

To address these challenges, numerous studies have explored sales forecasting methodologies, categorized into time series models and machine learning approaches. Time series models have been proven to be suited in handling linear data patterns, whereas machine learning performed better on nonlinear data. Given the nonlinear nature of our dataset, characterized by sales variations influenced by product discounts and customer behaviour, using machine learning to address these complexities becomes imperative. The paper proposes the application of XGBoost and TimesFM to enhance prediction accuracy and frequency, enabling daily sales forecasts.

1.3 Scope of study

1.3.1 The dataset in this study comprises daily sales transactions from a retail business, incorporating data from a total of 805 stores, including both online and offline channels.

1.3.2 Gathered from a total 5,349 products within 10 categories: apparel, bag, beauty, electronic, imported fashion, luxury fashion, luxury watches, owned brand fashion, personal care, and watches.

1.3.3 This dataset spans over five years, extending from January 2019 to December 2023.

1.3.4 Sales data within this research context are defined as net sales without VAT, encompassing sales after the deduction of discounts, trade GP, and Value Added Tax.

1.3.5 The scope of sales considered in this analysis is limited to those generated from finished goods, thereby excluding premium items and testers.

1.4 Expected or anticipated benefit gain

The implementation of sales prediction models offers substantial benefits that profoundly impact organizational performance.

1.4.1 Enhanced Decision-Making

Accurate sales prediction facilitates informed decision-making, normally business focus on profitability ratios relative to net sales. Profitability derives from revenue after deducting fixed and variable costs. Accurate sales forecasts allow for a proactive estimation of variable costs linked to sales. With precise sales, variable, and fixed cost data, management can assess business profitability realistically and adaptively to current market conditions.

1.4.2 Improved Resource Allocation

Sales prediction models enhance resource allocation efficiency by aligning operational resources with forecasted sales volumes. This optimization includes the allocation of personnel, such as in-store employees for offline operations or packers and logistics staff for online orders. Additionally, it ensures efficient scheduling of shipping trucks, packaging materials, computers, and other necessary devices.

1.4.3 Optimized Inventory Management

Accurate sales forecasts significantly improve inventory management by providing reliable demand projections. This capability enables companies to maintain optimal inventory levels, reducing the risks associated with overstocking and obsolescence. Enhanced inventory management also enhances operational efficiency and customer satisfaction by ensuring timely product availability. Past challenges in inventory management, such as warehouse overcapacity leading to product damage from structural failures, and the need for costly outsourced warehouse services.

In conclusion, the adoption of advanced sales prediction models represents a critical step toward achieving operational excellence in retail business, offering transformative benefits across decision-making, resource allocation, and inventory management processes.

Chapter 2

Literature Review

XGBoost was introduced as a powerful machine learning algorithm that excels in both speed and performance [1]. It's builds upon supervised machine learning, decision trees, and gradient boosting. Firstly, supervised machine learning employs algorithms to train a model, finding patterns in a dataset comprising features and corresponding labels. Subsequently, the trained model applies these insights to predict labels for new datasets based on their features. Secondly, decision trees construct models that predict labels by evaluating a hierarchical structure of if-then-else feature questions and estimating the minimum number of questions needed to assess the probability of making a correct decision. Decision trees can be used for both regression and classification tasks, wherein they forecast either categorical or continuous numeric values, respectively.

$$E = -\sum_{i=1}^{C} p_i \times \log_2(p_i) \quad (1)$$

Where C = the number of classes

p_i= the proportion of the i^{th} class within the set

Their evaluation is based on Entropy and Information Gain. Entropy (E) quantifies the impurity of a node, reflecting its heterogeneity according to the formula in Eq. 1. Information Gain is then calculated to achieve a concise and effective tree with superior classification accuracy by comparing the entropy post and pre-split on an attribute. The attribute yielding the highest information gain is selected, and the process iterates across each branch. Lastly, Gradient boosting is an ensemble learning process that combines predictions from several models into one. Models are built sequentially to correct the errors of the previous ones, giving more weight to predictors that perform better. This process minimizes the loss function using a gradient descent algorithm. Beginning with the fitting of the first model using the original data, subsequent models are fitted using the residuals of the previous ones. The outcome is a strong predictive model created through this iterative process as illustrated in Figure 1.

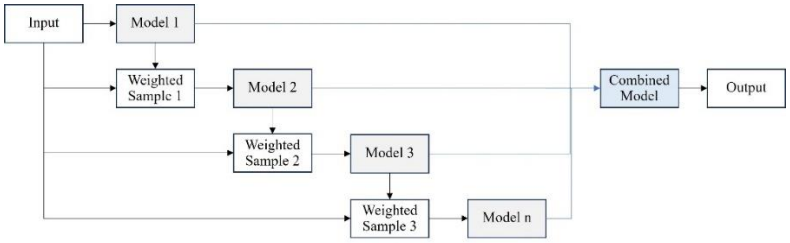

Figure 1: Gradient boosting ensemble learning process

Since XGBoost was introduced, it has been used for forecasting various topics through time. (Ji et al., 2019) [2] investigated sales forecasting using both Time Series Models (TSMs) and Machine Learning Algorithms (MLAs). Recognizing the limitations of TSMs, they proposed a combined approach using ARIMA and XGBoost models to capture both linear and nonlinear components of the data series, thereby overcoming the disadvantages of ARIMA because of their linear behaviour.

Similarly, (Wang and Guo, 2020) [3] applied a mixed model of ARIMA and XGBoost for stock price forecasting. Their study showed improvement over a single model, especially in datasets with partial nonlinear components which limit the performance of prediction in ARIMA. Integrating XGBoost helped explain nonlinear relationships and improve prediction accuracy by leveraging ensemble learning techniques. (Kalra et al., 2020) [4] compared XGBoost with tf-idf transform for predicting purchasing behaviour on Black Friday. While XGBoost showed promising performance, the study highlighted the challenge of overfitting and recommended preprocessing steps to remove noise in the dataset before applying the model. In line with this, (Xia et al., 2020) [5] demonstrated the advantage of XGBoost in predicting passenger car sales, enhancing accuracy through information gain and data correlation. They emphasized the importance of feature selection and proposed data filling algorithms to improve prediction accuracy, with mean filling outperforming other filling methods, and this approach will be adopted in this paper for the data cleansing process. Likewise, (Biswas et al., 2021) [6] evaluated five algorithms, including LSTM, XGBoost, Linear Regression, Moving Average, and Last Value model for stock market price prediction, with LSTM performing best for their purpose.

However, XGBoost ranked third in terms of performance, indicating its effectiveness but also room for improvement in specific contexts. (Fildes et al.,2022) [7] and (Pan, 2022) [8] explored retail forecasting and price prediction for BMW, respectively, both concluding that machine learning algorithms, including XGBoost, provided more accurate forecasts for nonlinear data compared to traditional methods. Thereby XGBoost given the best performance in Pan research. (Guliyev and Mustafayev, 2022) [9] predicted changes in WTI crude oil price using machine learning models, with XGBoost consistently outperforming other metrics. They attributed XGBoost's success to its ability to select the most significant features and effectively improve prediction accuracy. (Ganapathy et al., 2022) [10] Compared various algorithms for rainfall forecasting and found XGBoost to be the most accurate. According to enhancement from the conventional Gradient Boosting methodology, multiple decision trees have been assigned respective weights to explicitly give more importance to trees for determining the final output. Moreover, training time raking in the second, due to parallel learning process,

reduces the overall time for training by optimizing the underlying hardware. (Mitra A., Jain A., Kishore A., et al., 2022) [11] studied five regression techniques of machine learning, Random Forest (RF), XGBoost, Adaptive Boosting (AdaBoost), Artificial Neural Network (ANN), and hybrid (RF-XGBoost-LR) for demand forecasting in multi-channel retail company. As a result, the hybrid model received the highest accuracy, both RF and XGBoost jointly overcome the problem of overfitting and training error in linear regression, concerned of determining the relationships between dependent and independent variables. Furthermore, XGBoost minimum number of resources with parallel tree in ensemble method, rapid model exploration. Lastly, (Akanksha A., Yadav D., Jaiswal D., et al., 2022) [12] concluded that XGBoost is best performed for store-sales forecasting.

In conclusion XGBoost model has proven its ability on nonlinear data, outperforming in accuracy and efficiency across numerous research studies, making it a preferred choice for forecasting tasks in various domains.

Historically, time-series forecasting predominantly relied on statistical models like ARIMA and machine learning methods such as XGBoost. However, recent advancements in deep learning have catalysed the emergence of neural network-based approaches, offering improved accuracy and flexibility. (Abhimanyu D., Weihao K., Rajat S., Yichen Z., 2024) [13] introduced the TimesFM framework, the evolution of decoder-only models for time-series forecasting, inspired by Natural Language Processing (NLP). This novel approach utilizes a decoder-style attention mechanism pretrained with input patching on an extensive repository of time-series data comprising both real-world and synthetic datasets. Through empirical assessments across diverse, previously unexamined forecasting datasets, the model demonstrates its ability to accurately predict outcomes across multiple domains, forecasting horizons, and temporal granularities. They address challenges of dataset scarcity, the unpredictable nature of time-series data across different domains, and varying granularities through extensive corpus pretraining. This involves integrating a diverse range of public time-series data sources such as Google Trends, Wikipedia Pageview statistics, and synthetic time-series.

Research indicates that directly predicting the entire forecasting horizon yields better results than employing a multi-step autoregressive approach for long-term forecasting. However, in the context of zero-shot forecasting where the exact horizon length is unknown in advance, this direct prediction method is not feasible. Therefore, there is a need for a universal model capable of forecasting across variable horizon lengths. TimesFM addresses this challenge by incorporating patching, a technique previously popularized by the successful PatchTST model (Yuqi N., Nam H.N., et al., 2022) [14]. Figure 2 illustrates the architecture of the TimesFM

model and demonstrates how the Patching technique operates. Unlike predicting individual data points or the entire horizon length at once, TimesFM divides both the context and horizon into patches. Assuming a context length of size L and patches of size p, the input is segmented into $N = L/p$ input patches. Concurrently, output patches of size h are defined. By allowing $h > p$, the authors discovered that TimesFM can efficiently learn to forecast horizons of varying lengths with improved speed and accuracy. Initially, the model selects values for p (input patch size) and h (output patch size). It then divides the input into $N = L/p$ input patches. During the first decoding step:

a. The first patch undergoes processing by an Input Residual Block.
b. The resultant output is combined with a positional encoding vector.
c. This output is then fed into a stacked Transformer, where causal self-attention ensures each output token only attends to preceding input tokens.
d. Similarly, the output from step 3 proceeds through an Output Residual Block to generate the output patch, representing the predicted horizon. This output patch is compared against actual data to compute the loss.

Subsequently, in subsequent decoding steps, the model processes additional input patches to predict subsequent output patches. Importantly, in practice, all patches associated with a given input are generated within a single training mini batch. Based on these findings, TimesFM shows promise and warrants further exploration to improve accuracy in time-series forecasting.

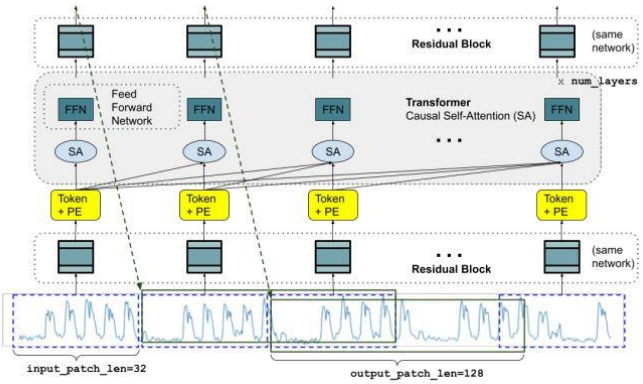

Figure 2: TimesFM model architecture

Chapter 3

Research Methodology

This study examines the predictive performance of the Original Method in comparison to the XGBoost and TimesFM models. The Original Method utilizes a weighted average approach, which entails aggregating data from the prior year to compute the daily average for forecasting current-day sales. Additionally, holidays are weighted at 1.25 times of the historical average, whereas weekdays are equal to an average weighting. The proposed new method will be evaluated in contrast to these approaches, following these outlined steps.

3.1 Data Preparation

The dataset in this study comprises daily sales transactions from a retail business, incorporating data from a total of 805 stores, including both online and offline channels. Gathered from a total 5,349 products within 10 categories: apparel, bag, beauty, electronic, imported fashion, luxury fashion, luxury watches, owned brand fashion, personal care, and watches. This dataset spans over five years, extending from January 2019 to December 2023. Sales data within this research context are defined as net sales without VAT, encompassing sales after the deduction of discounts, trade GP, and Value Added Tax. The scope of sales considered in this analysis is limited to those generated from finished goods, thereby excluding premium items and testers. The Extract Transform Load (ETL) process is facilitated through three environments: SAP High-performance Analytic Appliance On-premises (SAP HANA), SAP Data Intelligence Cloud (SAP DI), and the Google Cloud Platform (GCP) as illustrated in Figure 3.

Figure 3: Extract Transform Load (ETL) process

Initially, sales data are gathered from SAP HANA, sourced through two distinct points contingent upon the distribution channels: SAP Customer Activity Repository (SAP CAR) for consignment and owned shops and SAP Enterprise Resource Planning (SAP ERP) for credit and online sales. Input sales from SAP CAR originate

from scanning out at consignment locations via Mobile Inventory Management (MIM) or shops via Point of Sale (POS) systems, while credit and online sales are directly interfaced with SAP ERP via flat file.

Data stored within SAP systems can be linked to SAP DI using Cloud Connector, establishing a direct connection to the respective tables if data conversion is unnecessary. However, for complex datasets, transformation and modelling are necessitated through Core Data Service (CDS View) utilizing the Advanced Business Application Programming (ABAP) language. Subsequently, SAP DI serves as a facilitator between SAP HANA and GCP, akin to a bridge. Data pipelines are constructed within the modeler in SAP DI to effectuate data transformation utilizing available operators and Python scripts, subsequently exporting the transformed data to GCP in parquet format. This transformation process encompasses data cleansing, metadata-driven data type definition, and fundamental aggregation techniques aimed at ensuring data accuracy while minimizing data size.

Finally, data are transmitted from SAP DI to GCP and stored within Cloud Storage. Herein, complex calculations are conducted within BigQuery to transform and store the finalized data for running prediction models, leverage SQL, views, and stored procedures. During this transformation phase, the data is reconfigured into a time series dataset, with significant features derived from the structured grouping of product hierarchy and sales channels. Furthermore, mean filling techniques are applied to handle anomaly data, to ensure data integrity and consistency while also mitigating the risk of overfitting.

3.2 Data Aggregation and Data Cleansing

3.2.1 Product group hierarchy

The dataset records transactions individually, each identified by a unique barcode, mirroring the format of receipts printed for customers. Forecasting total sales for a brand based on individual barcode predictions presents challenges, mainly due to the potential for overfitting and subsequent inaccuracies in sales forecasts. Therefore, it is crucial to categorize products into distinct groups to address these challenges effectively.

The barcodes accompanying products exhibit varying formats depending on their respective principal suppliers. Consequently, it is necessary to map the barcode from each transaction to our article master database initially. Once mapped the barcodes to the article master, the article numbers are segmented into five ranges, which are then mapped each range to the product group hierarchy as illustrated with example in Figure 4.

Digits	Description	Example
1 2	Material type	Finished goods
3 4 5	Brand	Beauty
6 7	Article hierarchy level 1	Woman
8 9	Article hierarchy level 2	Face
10 11 12	Article hierarchy level 3	Skincare

Figure 4: Mapping of Product group hierarchy from the article number

Within each of the 10 categories, products will be grouped differently based on the characteristics of the brand. For example, the beauty category grouped at Article Hierarchy Level 3, the eight subcategories are classified into five groupings. Among these, four groups dominate in terms of sales volume and product type, while the remaining subcategories are categorized as "Others" due to their occasional and small sales volumes presented in Figure 5.

Article hierarchy level 3	Group
Bodycare	Bodycare
Make up	Make up
Service	Service
Skincare	Skincare
Others	Misc.
	Fragrance
	Accessories
	Haircare

Figure 5: Group of products in Beauty category

3.2.2 Distribution channels

With a total of 805 stores, averaging approximately 500 stores per brand, predicting sales at each store can introduce fluctuations compared to actual sales figures. Different sales channels also feature distinct pricing structures; typically, online channels offer higher discounts compared to offline sales through owned stores or consignment shops. Moreover, forecasting sales on a daily basis requires significant time investment. Therefore, to improve efficiency and accuracy, store types and sales channels are aggregated into online and offline categories during model training. The cleansing process begins by filling the distribution channel field found in the header of each document into each line within the same document. Subsequently, distribution channels are grouped according to their respective channel codes, as depicted in Figure 6.

System	Distribution Channel Code	Distribution Channel Description	Distribution Channel Group
SAP ERP	10	Credit	Offline
	11	Credit Promo Sales	
	13	Export	
	16	Retail Offline	
	18	Credit B2B	
	19	Credit B2B Promo	
	12	Online	Online
	14	Credit Online Wholesales	
	15	Credit Online Wholesales Promo	
	17	Retail Online	
SAP CAR	30	Retail Sales	Offline
	36	Retail Offline New sales channel	
	38	Retail B2B	
	32	Retail Online Department store	
	34	Retail Online Marketplace related	Online
	35	Retail Online Marketplace non related	
	37	Retail Online Brand.com	

Figure 6: Group of Distribution channel

However, the distribution channel depicted in Figure 6 pertains to 2023, whereas historical data spanning 2019 to 2022 are structured in legacy formats that do not distinguish between online and offline sales for shop and consignment store fulfillment. Fulfillment refers to instances where customers purchase goods online and the company ships them from a store or consignment location, recorded under distribution channel "30" denoting offline sales.

To correctly align historical data with the appropriate channels, transformations are applied based on the originating data systems. Initially, sales from shops are recorded in SAP ERP. Subsequently, additional data from the POS system is integrated, by grouping data according to shop code, sales date, and sloc (storage location). This aggregated data is then reconciled with SAP ERP records using

identifiers including customer code and sales date, then deducting the online sales amount from the total daily shop sales to determine the final figures for offline and online sales.

Similarly, sales from consignments recorded in SAP CAR necessitate integration with additional data from the WEB EDI system, specifically payment information received from department stores where consignment sales occurred. This process involves segregating online sales based on store code, sales date, and channel. After aggregating total consignment sales by store code and sales date, this data is reconciled with SAP records. Similarly, to shop data, the daily deduction of online sales amounts from total consignment sales yields the final figures for offline and online sales.

These transformations ensure that historical data aligns accurately with the respective online and offline sales channels as defined in the company's distribution strategies.

3.2.3 Transform the data into daily

In the context of the consignment channel, adherence to Thailand's tax regulations dictates that the point of sale occurs when goods are shipped to consignment locations. However, from our operational perspective, sales should be recognized when goods are sold to the end customer. To accommodate this regulatory requirement within our SAP system, we implement a bifurcated structure involving two distinct companies: Company 1000 serves as the master entity responsible for recording sales transactions to end customers, while Company 9000 represents the consignment company responsible for transactions with consignment locations.

Typically, at the beginning of each month, Company 9000 records sales agreed upon shipment, while Company 1000 begins the month with zero sales recorded. By the end of the month, Company 9000 reconciles its final sales figures and transfers this total to Company 1000. However, this practice leads to Company 1000 recognizing sales only on the last day of the month, which is not ideal for accurate forecasting. To ensures proper recognition of sales periods for accurate forecasting, sales from consignment locations using sales data from Company 9000 instead of Company 1000, by mapping profit centers and plant codes between both companies to facilitate precise forecasting.

3.2.4 Calculate net sales without VAT

According to the definition of sales in this paper, it refers to net sales excluding VAT, which means sales after deductions for discounts, trade gross profit, and VAT. Each transaction follows a distinct pricing structure set through discount codes in SAP categorized by article. The calculation of net sales without VAT primarily diverges into two methods, each aligned with specific systems: SAP ERP and

SAP CAR. These systems utilize different discount type codes, and their originating systems—POS for SAP ERP and MIM for SAP CAR—are tailored to collect amounts differently, in SAP ERP, discounts like Company Discount and Co Pro Discount include taxes, whereas in SAP CAR, they exclude taxes. Discounts are categorized into four groups: Company Discount, which is fully absorbed by the company; Co Pro Discount, shared with partners and absorbed based on contract percentages; Full Co Pro Discount, entirely absorbed by partners; and Trade GP, the amount deducted by department stores from total consignment sales as per contract terms. The signs in formulas vary depending on how values are recorded in the system, varying according to each discount type. Figures 7 and 8 illustrate the calculation process, where discounts are initially grouped, followed by the computation of net sales amounts excluding VAT.

System	Disc Type Code	Discount Group
SAP CAR	ZCH1	Company Discount
	ZCL1	
	ZCR1	Co Pro Discount
	ZCR4	
	ZCFA	Full Co Pro Discount
	ZCHA	
	ZAG1	Trade GP
	ZAG2	
	ZDS1	
	ZDS2	
	ZHGP	
	ZNGP	
	ZRNG	
	ZRS1	
	ZRS2	
	ZSGP	
SAP ERP	ZFD1	Company Discount
	ZFD2	
	ZHC1	
	ZHC2	
	ZHM1	
	ZHM2	
	ZHP1	
	ZHP2	
	ZHT1	
	ZHT2	
	ZSP1	
	ZSP2	
	ZFD3	
	ZFD4	
	ZHC3	
	ZHC4	
	ZHM3	
	ZHM4	
	ZHP3	
	ZHP4	
	ZHT3	
	ZHT4	
	ZSP3	
	ZSP4	
	ZCNC	
	ZCTN	
	ZCMC	Co Pro Discount
	ZNBN	
	ZCMV	
	ZNBV	
	ZCNV	
	ZCTV	
	ZCMF	Full Co Pro Discount
	ZNBF	

Figure 7: Group of discounts from discount type code

SAP CAR	
	Sales Amount (CAR)
-	Full Co Pro Discount
-	Company Discount
=	**Retail Sales**
÷	1.07
=	**Retail Sales excluded VAT**
+	Company Discount
+	Co Pro Discount
=	**Gross without Vat**
+	Trade GP
=	**Net Sales without VAT**

SAP ERP	
	Sales Amount (ERP)
-	Full Co Pro Discount
-	Company Discount
=	**Retail Sales**
÷	1.07
=	**Retail Sales excluded VAT**
+	Company Discount / 1.07
+	Co Pro Discount / **1.07**
=	**Gross without Vat**
=	**Net Sales without VAT**

Figure 8: Calculation for Net Sales without VAT segmented by systems

3.3 Apply XGBoost Regression

The dataset has been partitioned into training and testing sets to evaluate model performance. Subsequently, the XGBRegressor was implemented within the Python environment, incorporating essential input features, target variables, and specified parameters for training. These parameters notably include n_estimators, maximum tree depth (max_depth), learning rate (eta), and subsample. To provide clarity and facilitate reference, detailed definitions of the parameters utilized in this study are presented in Table 1.

Table 1: Definitions for parameters defined in XGBoost model

Parameters	Definitions
n_estimators	No. of trees in ensemble model.
max_depth	Maximum depth of a tree refers to the upper limit of nodes from the root to the farthest leaf.
eta	The learning rate used to weight each model, to prevents overfitting in each boosting step.
subsample	The number of samples training instances used in each boosting iteration to grow trees.

Parameter tuning for the XGBoost model, which involves adjusting the number of trees, maximum tree depth, learning rate, and subsample ratios, plays a crucial role in optimizing model performance. As the values of these parameters increase, the complexity of the model also escalates. However, maintaining a delicate balance between prediction accuracy, model complexity, and computational efficiency is essential. The time consumed in running the model directly impacts the speed of business decision-making. Additionally, excessively high parameter values can precipitate overfitting, thereby compromising result accuracy. The analysis emphasizes that each product category has its own characteristics and patterns. This

highlights the importance of customizing parameter settings to ensure the best performance of the model. All parameter specifics for each product category are summarized in Table 2. As mentioned in the data preparation phase, forecasting the total number of each category will involve sub-models based on product group hierarchy and distribution channels.

Table 2: Parameters configured in the model by product category

Product Categories	n_estimators	max_depth	eta	subsample
Apparel	200	8	0.1	0.3
Bag	150	9	0.1	0.3
Beauty	500	5	0.1	0.5
Electronic	700	9	0.1	0.5
Imported Fashion	250	9	0.1	0.3
Luxury Fashion	300	9	0.1	0.3
Luxury Watches	250	9	0.1	0.3
Owned Brand Fashion	300	5	0.1	0.5
Personal Care	200	9	0.1	0.3
Watches	500	8	0.1	0.5

3.4 Apply TimesFM

The prepared dataset includes a daily date column and unique columns that define each sub-model for categories based on product group hierarchy and distribution channels. The TimesFM model is subsequently applied within the Python environment, constructed with essential parameters detailed in Table 3. Analogous to XGBoost, the final prediction combines the outputs from all sub models.

Table 3: Definitions and configurations for parameters defined in TimesFM model.

Parameters	Definitions	Configurations
context_len	No. of trees in ensemble model. Multiple of 32 and not exceeding 512	320
horizon_len	Length of output patches during inferences	14
Input_patch_len	Length of input patches	32
output_patch_len	Length of output patches	128
num_layers	No. of transformer layers stacked on top of each other	20
model_dims	Model dimension of stacked transformer layer	1280
freq	Specify the frequency of the date column, for example "D" denotes daily and "M" denotes monthly.	D
num_jobs	Restrict the number of asynchronously forecasted series	1

3.5 Evaluation

The results are evaluated by Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) using the formulas as illustrated in Eq. 2 and Eq. 3 respectively.

$$\mathrm{MAE} = \left(\frac{1}{\mathrm{n}}\right) * \Sigma|y_i - x_i| \quad (2)$$

$$\mathrm{RMSE} = \sqrt{\left(\frac{1}{\mathrm{n}}\right) * \Sigma(y_i - x_i)^2} \quad (3)$$

Chapter 4

Results and Discussion

Figure 9 displays the sales forecasts generated by the TimesFM and XGBoost algorithms, presented in Million Baht. The predicted values from TimesFM are depicted by the green line, those from XGBoost by the blue line, and the actual sales numbers over time by the orange line. The alignment between predicted and actual figures across various product categories is evident, demonstrating the effectiveness of both TimesFM and XGBoost algorithms in accurately forecasting sales.

To provide a comprehensive comparison of outcomes, we will evaluate Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) in the following section. These metrics will further elucidate the accuracy and performance of the forecasting models depicted in Figure 9.

Figure 9: Comparison between sales prediction from XGBoost algorithm and actual value by product category

To facilitate a comparative analysis with the current prediction process, Table 4 presented a comparative analysis of the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) between the TimesFM model, XGBoost model and the Original Method across different product categories.

Table 4: Comparison of the MAE and RMSE between TimesFM, XGBoost and Original Method

Unit: Million Baht	**TimesFM**		**XGBoost**		**Original Method**	
Product Categories	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**
Apparel	0.06	0.08	0.13	0.15	0.15	0.20
Bag	0.13	0.22	0.39	0.45	0.43	0.56
Beauty	0.73	0.86	0.49	0.61	1.15	1.91
Electronic	0.69	0.83	1.52	2.44	1.71	2.77
Imported Fashion	0.48	0.80	0.73	0.90	0.79	1.06
Luxury Fashion	0.09	0.14	0.34	0.40	0.36	0.51
Luxury Watches	0.01	0.01	0.02	0.02	0.02	0.03
Owned Brand Fashion	0.37	0.50	0.46	0.63	0.86	1.43
Personal Care	0.91	0.94	1.21	2.00	0.89	1.50
Watches	0.36	0.48	1.39	1.79	2.01	2.54
Average	0.38	0.49	0.67	0.94	0.84	1.25

Chapter 5

Conclusion

In this study, accuracy, measured through Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), exhibited improvement across all product categories. In summary, the average MAE of the Original Method was 0.84, whereas the XGBoost model achieved a substantially lower value of 0.67, indicating an enhancement in accuracy of 20.24%. Similarly, for RMSE, the Original Method yielded an average score of 1.25, whereas the XGBoost model demonstrated a lower value of 0.94, reflecting an increase in accuracy by 24.80%.

Furthermore, this research underscores the efficacy of the TimesFM model in enhancing forecast accuracy when compared to the XGBoost model, improving in MAE and RMSE by 42.84% and 48.30%, respectively. TimeFMS emerges as a robust alternative that notably improves predictive capabilities, exhibiting an increase in MAE and RMSE by 54.40% and 61.12%, respectively, over the Original Method.

Moreover, the adoption both of TimesFM and XGBoost model facilitated a transition from monthly to daily prediction cycles, thus escalating operational efficiency for businesses, enabling them to refine their planning and marketing strategies for superior performance. Future research can further enhance the accuracy and efficiency of the model by integrating additional forecasting algorithms such as ARIMA, LSTM, or other models. By combining the strength of each model to overcome the predictive performance of the singular model. Therefore, explore the application of TimeFMS in different domains and its integration with other advanced forecasting techniques to optimize predictive accuracy even further.

REFERENCES

1. Chen, T. and C. Guestrin. *XGBoost: A scalable tree boosting system*. in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016.
2. Ji, S., et al., *An application of a three-stage XGboost-based model to sales forecasting of a cross-border e-commerce enterprise.* Mathematical Problems in Engineering, 2019.
3. Wang, Y. and Y. Guo, *Forecasting method of stock market volatility in time series data based on mixed model of ARIMA and XGBoost.* China Communications, 2020. **17(3)**: p. 205-221.
4. Kalra, S., et al. *Analysing and Predicting the purchases done on the day of Black Friday*. in *International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*. 2020.
5. Xia, Z., et al., *ForeXGBoost: passenger car sales prediction based on XGBoost.* Distributed and Parallel Databases, 2020. **38(3)**: p. 713-738.
6. Biswas, M., et al. *Predicting stock market price: A logical strategy using deep learning*. in *ISCAIE 2021 - IEEE 11th Symposium on Computer Applications and Industrial Electronics*. 2021.
7. Fildes, R., et al., *Retail forecasting: Research and practic-e.* International Journal of Forecasting, 2022. **38(4)**: p. 1283-1318.
8. Pan, L. and *Price Prediction for BMW Based on Multifactorial Linear and Machine Learning Model*. in *ACM International Conference Proceeding Series*. 2022.
9. Guliyev, H., et al., *Predicting the changes in the WTI crude oil price dynamics using machine learning models.* Resources Policy, 2022. **77**.
10. Ganapathy, G.P., et al., *Rainfall Forecasting Using Machine Learning Algorithms for Localized Events.* Computers, Materials and Continua, 2022. **71(2)**: p. 6333-6350.
11. Mitra, A., et al., *A Comparative Study of Demand Forecasting Models for a Multi-Channel Retail Company: A Novel Hybrid Machine Learning Approach.* Operations Research Forum, 2022. **3(4)**.
12. Akanksha, A., et al., *Store-sales Forecasting Model to Determine Inventory Stock Levels using Machine Learning*, in *5th International Conference on Inventive Computation Technologies (ICICT)*. 2022. p. 339-344.
13. Abhimanyu, D., et al. *A decoder-only foundation model for time-series forecasting*. in *International Conference on Machine Learning*. 2024.
14. Yuqi, N., et al. *A time series is worth 64 words: Long-term forecasting with transformers*. in *International conference on learning representations*. 2022.

VITA

NAME	Prathana Dankorpho
DATE OF BIRTH	17 May 1994
PLACE OF BIRTH	Bangkok
INSTITUTIONS ATTENDED	Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University
HOME ADDRESS	99/138 Rattanatibeth Road, Bangraknoi, Nonthaburi 11000
PUBLICATION	Dankorpho, P. (2024). Sales Forecasting for Retail Business using XGBoost Algorithm. Journal of Computer Science and Technology Studies, 6(2), 136–141. https://doi.org/10.32996/jcsts.2024.6.2.15